END-TO-END LANGUAGE RECOGNITION USING ATTENTION BASED HIERARCHICAL GATED RECURRENT UNIT MODELS

Bharat Padi[1*], *Anand Mohan*[2], *Sriram Ganapathy*[2]

[1]minds.ai, Bengaluru, India.
[2]Learning and Extraction of Acoustic Patterns (LEAP) Lab, Electrical Engineering,
Indian Institute of Science, Bengaluru, India

ABSTRACT

The task of automatic language identification (LID) involving multiple dialects of the same language family on short speech recordings is a challenging problem. This can be further complicated for short-duration audio snippets in the presence of noise sources. In these scenarios, the identity of the language/dialect may be reliably present only in parts of the speech embedded in the temporal sequence. The conventional approaches to LID (and for speaker recognition) ignore the sequence information by extracting long-term statistical summary of the recording assuming an independence of the feature frames. In this paper, we propose to develop an end-to-end neural network framework utilizing short-sequence information in language recognition. A hierarchical gated recurrent unit (HGRU) model with attention module is proposed for incorporating relevance in language recognition, where parts of speech data are weighted more based on their relevance for the language recognition task. Experiments are performed using the language recognition task in NIST LRE 2017 Challenge using clean, noisy and multi-speaker speech data. In these experiments, the proposed approach yields significant improvements over the conventional i-vector based language recognition approaches as well as previously proposed approach to language recognition using recurrent networks.

Index Terms: End to end language identification, hierarchical GRU, attention.

1. INTRODUCTION

The problem of recognizing the spoken language of a given audio segment is of considerable interest for several commercial applications like speech translation [1], multi-lingual speech recognition [2], document retrieval [3] as well as in defense and surveillance applications [4]. In the recent years, several advances in signal processing, machine learning and the application of factor analysis methods have contributed to improving the performance of language recognition systems [5]. However, the task can be challenging when the recognition involves multiple dialects of the same language and for cases in which the test audio segments are short in duration, especially in the presence of noise and other artifacts. In this paper, we propose a modeling framework to address some of the challenges in LID systems.

Traditionally, phoneme recognition followed by language modeling (PRLM) was one of the popular methods for automatic LID task [6, 7]. This approach uses a multilingual phoneme recognizer to generate phoneme sequences which are converted to language model (n-gram) features for the LID classifier. In the recent past, the use of deep neural network (DNN) based posterior features were attempted for LID [8] and the use of bottleneck features derived from a speech recognition acoustic model have recently shown consistent improvements for language recognition [9, 10].

The development of i-vectors as one of the primary features for LID was first introduced in [11]. They are features of fixed dimensions derived from variable length speech utterances using a background model [12] and capture long term information of the speech signal such as speaker and language. The i-vectors extracted from the training data are then used to train classifiers such as support vector machines (SVMs) [13, 14].

End-to-end approaches to language recognition have been explored with long short term memory (LSTM) networks [15, 16, 17] and DNNs [18]. Both [17, 18] use attention based approaches. The work in [17] uses LSTM architecture with attention mechanism for short duration (3s) utterances. A recent approach using curriculum learning is also explored for noise robust language recognition [19]. The neural network based models tend to perform well only for short duration audio segments. Hence, the state-of-art language recognition systems using large scale NIST language recognition evaluation (LRE) challenges, continue to use the i-vector based approaches [20].

In this paper, we propose an end-to-end approach for language recognition using a hierarchical gated recurrent unit (HGRU) architecture with attention module. The HGRU model consists of two layers of GRU followed by a bidirectional GRU layer and it implements a temporal hierarchy where the initial layer accumulates local information from 100msec segments and the next layer accumulates information from segments of 1sec length. The output of representations at 1sec segment level are fed to the attention model [21] which is then mapped to the language classes.

Experiments are performed using the training and evaluation data of NIST Language Recognition Evaluation (LRE) 2017 challenge in the fixed training condition. In the LRE experiments, the end-to-end HGRU approach provides improved language recognition performance compared to the traditional i-vector baseline as well as the previously proposed LSTM model for LID [15, 16]. The HGRU model is also shown to be significantly faster in terms of computational complexity. Further, additional experiments are performed with various noise conditions and with multi-speaker data. In these experiments, the HGRU model shows significant benefits over the baseline approach.

The rest of the paper is organized as follows. In Section 2, we describe the proposed HGRU based LID system. The experimen-

This work was funded partly by grants from the Department of Science and Technology (DST) Early Career Award (ECR01341) and the Pratiksha Young Investigator Award.

* This work was performed when the first author was working in the LEAP lab, Indian Institute of Science.

tal setup used for the LRE2017 dataset as well as the details of the baseline system are provided in Section 3. The results of various LID experiments are reported in Section 4 which is followed by a discussion on the proposed model in Section 5. The summary of the work is provided in Section 6.

2. HGRU BASED LID SYSTEM

Long Short-Term Memory (LSTM) recurrent neural networks (RNN) [22] were proposed to overcome the difficulty of handling long term dependencies in the input sequences by vanilla RNNs. A simplified version of the LSTM function is the Gated Recurrent Unit (GRU) proposed in [23]. In many tasks, the GRUs that have a smaller number of parameters are shown to achieve or improve over the performance of LSTMs [24]. Even with LSTM/GRU models, the modeling of long sequences can be cumbersome [16] as the sequences in LID can be of duration 10sec (1000 frames at 100 Hz sampling) and 30sec (3000 frames) or even longer. In order to model such long sequences, we propose a novel hierarchical bidirectional GRU network with attention in this paper.

The block schematic of the proposed model is given in Fig. 1. The input to the model is a sequence of acoustic bottleneck (BN) features from a previously trained deep neural network automatic speech recognition (ASR) system (similar to the baseline model [20]), where each feature vector represents information from short window (25msec with a hop of 10msec) of the speech utterance. At the first layer, a 256 cell unidirectional GRU block accumulates information across a window of 200msec i.e, a sequence of 20 feature vectors with a shift of 100msec over the entire input sequence. The output from the first layer is a sequence of vectors that are sampled every 100msec with each vector representing information from overlapping 200msec segments of input speech. This is then fed to the second layer of GRU block with 512 cells where the information is accumulated over a window of 1sec (i.e., 10 vectors from the previous layer sampled every 100msec.) The accumulated 1sec vectors from second layer are fed to the final bidirectional GRU layer [25] with 512 cells in the third layer. The forward hidden state $\overrightarrow{\mathbf{h}_t}$ and the backward hidden state $\overleftarrow{\mathbf{h}_t}$ of bidirectional GRU are then concatenated and used in the attention network i.e, $\mathbf{h}_t = [\overrightarrow{\mathbf{h}_t}, \overleftarrow{\mathbf{h}_t}]$.

The output of the three layer hierarchical GRU model contains representations at 1s level. Instead of directly accumulating the statistics, we propose to use an attention model to weight the 1sec representations based on their relevance to the language classification task. The attention method [21] provides an efficient way to aggregate the sequence of 1sec vectors. The attention mechanism used in this work is shown in Fig. 2. The model implements the following set of equations,

$$\mathbf{u}_t = tanh(\mathbf{W}_l \mathbf{h}_t + \mathbf{b}_l) \tag{1}$$

$$a_t = \frac{exp(\mathbf{u}_t^T \mathbf{u}_l)}{\sum_t exp(\mathbf{u}_t^T \mathbf{u}_l)} \tag{2}$$

$$\boldsymbol{l} = \sum_t a_t \mathbf{h}_t \tag{3}$$

Here, $\mathbf{W}_l$, $\mathbf{b}_l$ are the weights and the bias of the attention module which are learned in training process along with the vector $\mathbf{u}_l$. $\boldsymbol{l}$ denotes the fixed dimensional embedding from the input sequence. The attention module based on the similarity of $\mathbf{u}_t$ with $\mathbf{u}_l$ assigns normalized weights $\mathbf{a}_t$ using a softmax function. These weights are then used for aggregating the output sequence of bidirectional GRU layer to the utterance level representation $\boldsymbol{l}$, which is then mapped to

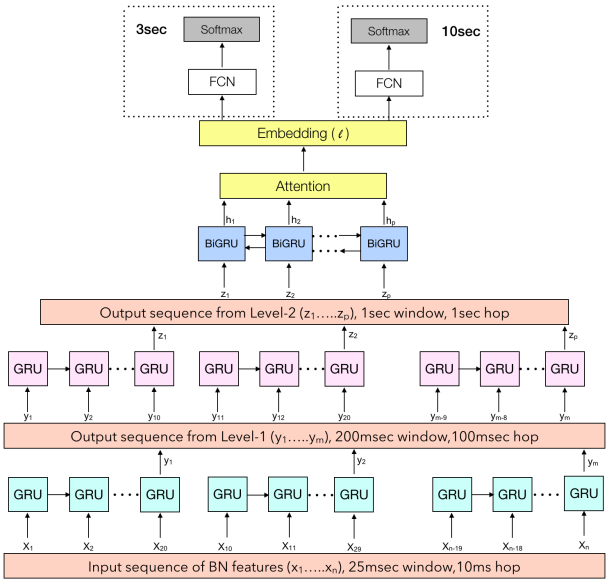

Fig. 1. End-to-end Hierarchical GRU RNN with attention module and duration dependent target layers.

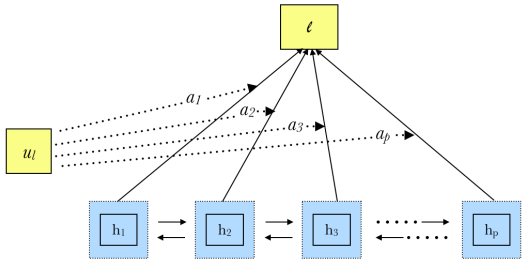

Fig. 2. Attention Mechanism in HGRU.

the final language targets through a layer of fully connected network (FCN). Since the distribution of the embeddings $\boldsymbol{l}$ are quite different for short and long duration inputs, we use two separate target layers, one for short duration inputs that are of the order of 3sec and other one with longer duration input sequences that are 10sec or above. The entire network is trained using Adam optimization and Back Propagation Through Time (BPTT) algorithm [26].

3. EXPERIMENTAL SETUP

3.1. Data Used

The LID system training is performed on the LRE2017 training LDC2017E22 dataset and the evaluation is performed using LRE2017 evaluation setup. The LRE2017 training data has five major language clusters (Arabic, Chinese, English, Slavic and Iberian) with 14 target dialects with a total duration of 2069 hours in 16205 files. The development dataset consists of 3661 files which contain 253 hours of audio and the evaluation dataset consists of 25451 files with 1065 hours of audio. The development and evaluation datasets are further partitioned into utterances of duration 3sec, 10sec or 30sec and the audio extracted from video data consisting of 1000sec recordings.Since the LRE2017 was a closed language set LID evaluation, we use the accuracy as the primary metric for evaluating various models in this work. We also report the official LRE cost

metric $Cavg$ for reference on the original LRE evaluation set. It is worth noting that the cost metric of $Cavg$ can be improved with score calibration using a development set. In this paper, we have reported LID results from raw scores without any calibration and this may be somewhat disadvantageous to neural network models as the neural network scores estimate multi-class posterior probabilities while the SVM estimates the scores in a one-versus-rest classification framework.

The robustness of the proposed model towards noise is evaluated by experimenting with a noisy version of the eval data. We use five different types of noise (Babble, Restaurant, Airport, Street, Subway) at various signal-to-noise ratios (0, 5, 10, 15 and 20 dB).The noise is either added to the entire input utterance or only to one half of the input utterance (to simulate non-stationary noise effects). We also evaluate multi-speaker conditions where the evaluation data of the same language from 2 speakers is merged to form a single utterance. These scenarios reflect various practical conditions involving stationary, non-stationary noises as well as conditions with multiple talkers of the same language.

3.2. Feature Extraction

We extract 80 dimensional bottleneck features (BNF) from a DNN trained for automatic speech recognition using Kaldi [27] framework. For the DNN Bottleneck (BN) feature extraction, we trained the model using 39 ($13+\Delta+\Delta\Delta$) dimensional MFCC features with 10ms frame rate over 25ms windows on labeled speech data from Switchboard SWB1 and Fisher corpora ($\sim$2000 hours). The model uses 7 hidden layers with ReLU activation and layer-wise batch normalization.

Once the BNF features are extracted for the LID data, a speech activity detection (SAD) algorithm was applied to remove the unvoiced frames [28]. This was followed by cepstral mean variance normalization (CMVN) done over each utterance on the BNF features, followed by a sliding window cepstral mean variance normalization (CMVN) over a 3sec window.

3.3. Baseline System

3.3.1. i-vector LDA-SVM

The i-vectors are one of the most widely used features for language recognition and we follow the procedure described in [11] for their extraction. They are features of fixed dimension derived from a variable length sequence of front end BN features. A Gaussian Mixture Universal Background Model (GMM-UBM) with 2048 mixtures is trained by pooling the features (BNF) from all the utterances in the training dataset. The means of the GMM are adapted to each utterance using the Baum-Welch (BW) statistics of the front-end features and a Total Variability Model (TVM) is trained using the BW statistics to derive a total variability subspace of 500 dimensions. The i-vectors, once extracted for each of the speech files, are then processed with length normalization and linear discriminant analysis (LDA). A support vector machine (SVM) classifier is then trained on these i-vectors and used for language identification [20].

3.3.2. LSTM

In [15, 16], the authors have proposed a Long Short Term Memory Recurrent Neural Network (LSTM-RNN) based end-to-end model for exploiting temporal information for LID. In [16], for experiments on NIST datasets, it is shown that even though the LSTM model outperforms i-vector baseline on short duration (3sec) test segments, its performance is relatively poor on longer duration test segments (10sec, 30sec). We implement their best performing LSTM model, which is a two layer LSTM with 512 units in each layer followed by an output softmax layer as a baseline end to end system.

Table 1. LRE2017 evaluation results on clean evaluation data in terms of accuracy in % (and Cavg in parenthesis) for baseline system [20], LSTM model [16] and the proposed HGRU model.

Dur. (sec)	ivec [20]	LSTM [16]	HGRU
3	53.8 (0.53)	54.7 (0.55)	**55.1** (0.55)
10	72.3 (0.27)	72.1 (0.35)	**74.1** (0.32)
30	83.0 (0.13)	76.1 (0.28)	**83.0** (0.23)
1000	**56.2** (0.54)	42.8 (0.79)	53.5 (0.62)
overall	67.9 (0.37)	64.3 (0.48)	**68.5** (0.42)

Table 2. LRE2017 evaluation results in terms of accuracy (%) for noisy data and partial noisy data for i-vector baseline system [20] and the proposed HGRU model.

Cond.	ivec [20]	HGRU
Clean	72.3	**74.1**
Noisy 5 dB	**47.9**	45.8
Noisy 10 dB	53.8	**53.9**
Noisy 15 dB	57.8	**60.2**
Noisy 20 dB	60.0	**64.2**
Avg.	54.9	**56.0**
Partial Noisy 5 dB	53.3	**55.3**
Partial Noisy 10 dB	55.8	**60.2**
Partial Noisy 15 dB	58.5	**63.2**
Partial Noisy 20 dB	59.8	**65.7**
Avg.	56.9	**61.1**

4. RESULTS

The results of LRE evaluation on various conditions are reported in Table 1. The LRE evaluations of 3sec, 10sec and 30sec use clean recordings while the 1000sec recordings use audio extracted from video data. Note that none of the models were trained with any audio extracted from video data and all the models perform similar or worse than 3sec condition. The LSTM model that is previously proposed for end-to-end language recognition [15, 16] performs better than the baseline i-vector system in terms of accuracy for the 3sec condition. However the performance drops below the baseline system for longer duration conditions as reported in [16]. In particular, the accuracy drops significantly compared to the baseline i-vector system for 30sec condition and the 1000sec condition.

For the case of HGRU model, the performance of the LID system improves over the baseline system for input duration of 3sec, and 10sec, while being comparable to the baseline system for the 30sec. This also marks a significant improvement for the HGRU system over the previous end-to-end baseline using LSTM model. For the 1000sec condition, the performance of the HGRU system is worse than the baseline system however being much better than the LSTM model. These experiments highlight that the HGRU with attention provides a considerable advancement for end-to-end LID task.

The results for the LID experiments on the noisy data are shown in Table 2. Here, we use the 10sec evaluation data and report the results for two separate conditions, one in which the noise is uniformly added to the input utterance and the second one in which the noise is

Table 3. Approximate computational time in seconds for ten 30sec eval files using a single CPU. Machine Specification: 32 CPU, 8 core, 2 thread Intel x86_64 machine with 16 GB Nvidia Quadro P5000 GPU cards.

	ivec [20]	LSTM [16]	HGRU
CPU	12	51	8
GPU	12	11.5	1.5

selectively added only to the first half of input utterance. The second scenario simulates a more common condition in practical applications as it creates a non-stationary noise environment encountered in real world noise. The SNR value reported is SNR for the first half of the utterance where the noise is added (and not the average SNR of the entire recording). As seen in these experiments, the proposed HGRU model provides significant improvements over the baseline system for both the noisy condition (for all SNR values except the 5 dB condition) as well as the partial noisy condition (for all SNR values). The relative improvements over the baseline system are about 10% for the HGRU model for the partial noise condition.

5. DISCUSSION

5.1. Computational Complexity

During the testing phase, the end-to-end HGRU model involves fewer number of steps relative to an i-vector based system since it directly uses the front end BN features. We perform a comparison of the computational complexity between the i-vector baseline system, LSTM baseline system and the proposed model in terms of running time on a single CPU based system for LID score generation. This run time computation was performed using 30sec test files. On the average, the i-vector baseline system requires 1.2sec of computation time, while the HGRU system requires only 0.8sec(0.15sec) on CPU(GPU). This is a noticeable improvement in the computational complexity of the HGRU system which is achieved at comparable or improved LID performance (Table 1). It is also worthwhile noting that the HGRU system has lower computational complexity than the LSTM system owing to its architecture along with significant improvement in performance.

5.2. Attention Analysis

In this subsection, we analyze the role of the attention mechanism in the proposed HGRU model. We plot the spectrogram of a partially corrupted speech recording (first 5sec at 10 dB SNR) and the corresponding attention vector which is computed at 1sec resolution in Fig. 3. As can be seen, the attention weights for the later part of the utterance where the SNR is high are relatively higher making them more relevant to the task. In our informal analysis, we have also found that even in clean data conditions, the attention vector makes intuitive sense (for example, in classifying British English recordings, the attention vector tends to provide higher weight for 1sec regions containing more accented speech segments).

5.3. Multi-talker LID and LID without SAD

We also performed two additional LID experiments with the proposed HGRU model. The first experiment uses speech recordings in testing that contain multiple speakers. This is obtained by merging 3sec speech utterances in the clean LRE evaluation set from multiple talkers of the same language. The second experiment explores the

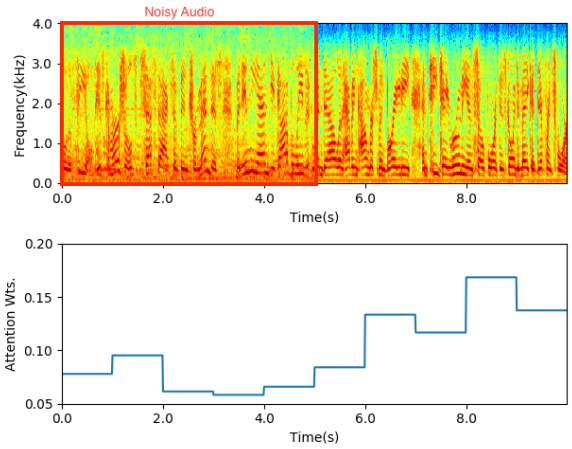

Fig. 3. Sample Spectrogram of a partially noised utterance with attention weighting from HGRU. Noise (10 dB SNR) was added to the first 5sec of the utterance.

Table 4. LID accuracy in % for additional experiments with multiple speakers speaking the same language and the experiments without any SAD information.

Cond.	ivec [20]	HGRU
Multi-Speaker	60.6	**67.7**
Without SAD information	49.7	**52.7**

sensitivity of the LID systems to the absence of any speech activity detection (SAD) information. Both these experiments use the 3sec recording data from the LRE evaluation setup. As seen in Table 4, the proposed HGRU model is more robust to the presence of multiple talkers in the evaluation dataset. The HGRU model is also less sensitive to the absence of any SAD information for the 3sec audio snippets. These experiments confirm that the HGRU model is able to efficiently model the time series for the language classification by relevance weighting based on the attention mechanism.

6. SUMMARY

The following are the novel contributions from the current work

- A new hierarchical gated recurrent unit (HGRU) based LID system is proposed for end-to-end spoken language recognition that also contains an attention mechanism for relevance weighting.
- The proposed HGRU model is shown to significantly improve over the previous attempts for end-to-end LSTM based language recognition systems.
- The HGRU approach is robust to the presence of noise in the test data as well as in non-stationary conditions like partially corrupted speech data or multi-talker speech segments.
- The attention mechanism in HGRU plays the role of relevance weighting, where portions of the speech signal that are more relevant to classification decision are taken into account. The conventional system based on i-vectors ignores sequential speech information by computing a statistical average.

From this work, we find that the research direction of using temporal sequence information along with attention based relevance weighting is promising and warrants further exploration in the future for large scale speaker and language recognition tasks.

7. REFERENCES

[1] Alex Waibel, Petra Geutner, L Mayfield Tomokiyo, Tanja Schultz, and Monika Woszczyna, "Multilinguality in speech and spoken language systems," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1297–1313, 2000.

[2] Tanja Schultz and Alex Waibel, "Language-independent and language-adaptive acoustic modeling for speech recognition," *Speech Communication*, vol. 35, no. 1-2, pp. 31–51, 2001.

[3] Ciprian Chelba, Timothy J Hazen, and Murat Saraclar, "Retrieval and browsing of spoken content," *IEEE Signal Processing Magazine*, vol. 25, no. 3, 2008.

[4] Kevin Walker and Stephanie Strassel, "The RATS radio traffic collection system," in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[5] Haizhou Li, Bin Ma, and Kong Aik Lee, "Spoken language recognition: from fundamentals to practice," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1136–1159, 2013.

[6] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, pp. 31, 1996.

[7] Jiri Navratil, "Spoken language recognition - A step toward multilinguality in speech processing," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 6, pp. 678–685, 2001.

[8] Mhamed Faouzi BenZeghiba, Jean-Luc Gauvain, and Lori Lamel, "Phonotactic language recognition using MLP features.," in *Interspeech*, 2012.

[9] Fred Richardson, Douglas Reynolds, and Najim Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1671–1675, 2015.

[10] Yun Lei, Nicolas Scheffer, Luciana Ferrer, and Mitchell McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1695–1699.

[11] Najim Dehak, Pedro A Torres-Carrasquillo, Douglas Reynolds, and Reda Dehak, "Language recognition via i-vectors and dimensionality reduction," in *Interspeech*, 2011.

[12] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[13] Sriram Ganapathy, Kyu Han, Samuel Thomas, Mohamed Omar, Maarten Van Segbroeck, and Shrikanth S Narayanan, "Robust language identification using convolutional neural network features," in *Interspeech*, 2014.

[14] Bharat Padi, Shreyas Ramoji, Vaishnavi Yeruva, Satish Kumar, and Sriram Ganapathy, "The LEAP language recognition system for lre 2017 challenge-improvements and error analysis," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 31–38.

[15] Javier Gonzalez-Dominguez, Ignacio Lopez-Moreno, Hasim Sak, Joaquín González-Rodríguez, and Pedro J. Moreno, "Automatic language identification using long short-term memory recurrent neural networks," in *INTERSPEECH*, 2014.

[16] Ruben Zazo, Alicia Lozano-Diez, and Joaquin Gonzalez-Rodriguez, "Evaluation of an LSTM-RNN system in different nist language recognition frameworks," in *Proc. of Odyssey 2016 Speaker and Language Recognition Workshop*. ATVS-UAM, June 2016.

[17] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, and Bo Xu, "End-to-end language identification using attention-based recurrent neural networks.," in *INTERSPEECH*. ISCA, 2016, pp. 2944–2948.

[18] KV Mounika, Sivanand Achanta, HR Lakshmi, Suryakanth V Gangashetty, and Anil Kumar Vuppala, "An investigation of deep neural network architectures for language recognition in indian languages.," in *INTERSPEECH*, 2016, pp. 2930–2933.

[19] Ravi Kumar Vuddagiri, Hari Krishna Vydana, and Anil Kumar Vuppala, "Curriculum learning based approach for noise robust language identification using DNN with attention," *Expert Systems with Applications*, 2018.

[20] Seyed Omid Sadjadi et al., "The 2017 NIST language recognition evaluation," in *Proc. Odyssey*, Les Sables dÓlonne, France, June 2018.

[21] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[22] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[24] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

[25] Mike Schuster, Kuldip K. Paliwal, and A. General, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, 1997.

[26] Paul J Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[27] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[28] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, pp. 1–3, 1999.